CS534 - MACHINE LEARNING

Author: Vy Bui
Email: buivy@oregonstate.edu

The School of Electrical Engineering and Computer Science
Oregon State University

Contents

Chapter 1

SPAM PREDICTOR

Chapter 2

LINEAR REGRESSION

Chapter 3

LOGISTIC REGRESSION

Chapter 4

FEATURE SELECTION

Chapter 5

PERCEPTRON

A perceptron is a binary classifier using the Heaviside step function instead of sigmoid as in Logistic Regression. Because this function is not differentiable, gradient descent cannot be used, and the perceptron learning algorithm is used instead.

5.1 Algorithm Learning Algorithm

The algorithm starts with random weights, iteratively updates them whenever the model makes a prediction mistake.

$$w_{t+1} = w_t - \eta_t(\hat{y}_n - y_n)x_n$$

Intuitively, if $y_n = 1$ and $\hat{y}_n = 0$, we have $w_{t+1} = w_t + x_n$, and if $y_n = 0$ and $\hat{y}_n = 1$, we have $w_{t+1} = w_t - x_n$.

5.2 Perceptron/hinge loss

Perceptron loss

$$L_P(w) = \frac{1}{N}\sum_{i=1}^{N} max(0, -y_i \mathbf{w}^T \mathbf{x}_i)$$

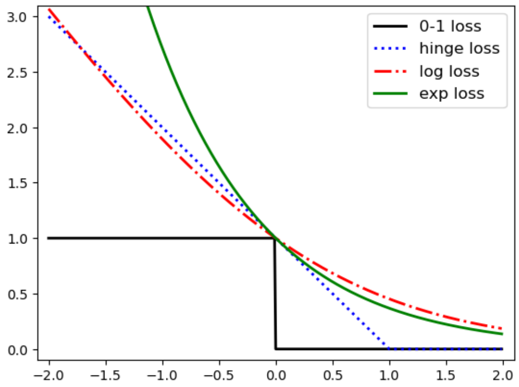

Figure 5.1: Various loss functions

5.3 Sub-gradient descent

The Subgradients of a convex function is a way of generalizing the notion of a derivative to work with functions which have local discontinuities. For a convex function of several variables, $f : R^n \to R^n$, we say that $g \in R^n$ is a **subgradient** of f at $\mathbf{x} \in dom(f)$ if for all vectors $z \in dom(f)$

$$f(z) \geq f(\mathbf{x}) + \mathbf{g}^T(\mathbf{z} - \mathbf{x})$$

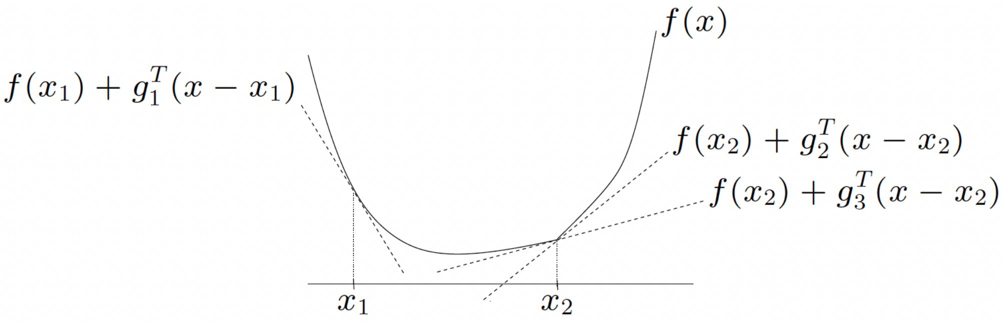
$f(x)$
$f(x_1) + g_1^T(x - x_1)$
$f(x_2) + g_2^T(x - x_2)$
$f(x_2) + g_3^T(x - x_2)$
x_1
x_2

Figure 5.2: Illustration of subgradients

5.4 Perceptron Convergence Theorem

If data is **linearly separable**, perceptron algorithm will find a linear classifier that classifies all data correctly in at most $O(r^2/\gamma^2)$, where $r = max||X_i||$ is **the radius of data** and γ is the **maximum margin**.

Margin is the min distance from data points to the decision boundary. The bigger the margin is, the easier the classification problem is. Also, the model will converge faster.

5.5

5.6

Voted and Average Perceptrons Structured Perceptrons

Chapter 6

KERNEL METHODS

By definition, kernels are functions $k : X \times X \to \mathbb{R}$ for which there exists a Hilbert space $\mathbb{H}$ and a feature map $\phi : X \to \mathbb{H}$ such that

$$k(x, x') = \phi(x)^T \phi(x')$$

Intuitively speaking, kernels calculate the relationships between every pair of observations. And these observations are used to find the **support vector classifier**.

The kernel trick helps calculate the relationships between every pair as if they are in the higher dimensions but they actually don't do the transformation. This reduces the amount of computation significantly.

Inner product? Hilbert space?

Design matrix Φ

dual representation

Gram matrix $K = \Phi\Phi^T$

positive semidefinite matrix

Chapter 7

SVM

SVM?

Linear SVM?

Non-linear SVM?

Hard/Maximum margin SVM issues
- Has no solution for nonlinearly separable data
- Overfits to outliers

Soft margin
- Introduce slack to the hard margin constraints (allow misclassification, bias/variance tradeoff)
- Minimizing L2 Regularized
- Data points for which $\epsilon = 0$ are correctly classified and are either on the margin or on the correct side of the margin. Points for which $0 < \epsilon \leq 1$ lie inside the margin, but on the correct side of the decision boundary, and those data points for which $\epsilon > 1$ lie on the wrong side of the decision boundary and are misclassified. - Support Vectors include Margin Support Vectors ($\epsilon = 0$) and Non-margin Support Vectors ($\epsilon > 0$ and $0 < \epsilon < 1$ and $\epsilon > 1$)

L_2 SVM?

Chapter 8

NAIVE BAYES

Chapter 9

DECISION TREE

9.1 Mutual Information

Entropy

$$H(X) = -\sum_{k=1}^{K} p(X=k) log_2 p(X=k) \tag{9.1}$$

Entropy for Binary Random Variable

$$H(X) = -p(X=1) log_2 p(X=1) - p(X=0) log_2 p(X=0) \tag{9.2}$$

Joint Entropy

$$H(X,Y) = -\sum_{x,y} p(x,y) log_2 p(x,y) \tag{9.3}$$

Conditional Entropy

$$H(Y|X) = H(X,Y) - H(X) = -\sum_{x,y} p(x,y) log_2 p(x,y) - \sum_{x} p(x) log_2 \frac{1}{p(x)} \tag{9.4}$$

or more general

$$H((X_1, X_2, ..., X_n)) = \sum_{i=1}^{n} H(X_i|X_1, ..., X_{i-1}) \tag{9.5}$$

(9.4)
Mutual information tells us how similar two distributions are.

$$I(X,Y) = H(X) - H(X|Y) = H(Y) - H(Y|X) \tag{9.6}$$

[Mur22], 6.3

Notes from [Mur22], 18

- Multi-way split might cause **data fragmentation** (too little data might fall into each subtree), resulting in overfitting.
-
-

[HTF17], 9.2

Chapter 10

Emsemble Methods

10.1 Bagging

refer to [Mur22], chapter 18.3 Bagging.

10.2 Boosting

[Bis06], 14.3

Chapter 11

Unsupervised Learning

11.1 Clustering

For **Hierarchical Clustering**, refer to [Mur22], chapter 21.2 Hierarchical clustering.

For **Flat Clustering**, refer to [Mur22], chapter 21.3 K Means Clustering.

Chapter 12

Dimensionality Reduction

12.1 Motivation

Many problems involve a huge number of features, which make the training more expensive and make it harder to search for a good solution. This problem is known as the *curse of dimensionality.* Oftentimes, reducing the number of features can

- speed up training
- filter out noise and unnecessary details, thus result in higher performance models.
- useful for data visualization

[Ger19]
Furthermore, in high-dimensional data sets, data points are likely to be far from each other (sparse).

12.2 Prerequisite

Linear Algebra: Eigenvectors Statistics: covariance

Bibliography

[Bis06] Christopher M. Bishop. *Pattern Recognition and Machine Learning.* Springer, 2006.

[Ger19] Aurelien Geron. *Hands-on Machine Learning with Scikit-learn, Keras, and Tensorflow.* O'Reilly, 2019.

[HTF17] Trevor Hastie, Robert Tibshirani, and Jerome Friedman. *The Elements of Statistical Learning.* Springer, 2017.

[Mur22] Kevin Patrick Murphy. *Probabilistic Machine Learning, An Introduction.* The MIT Press, 2022.